Predicting Software Fault Content and Fault Location

Principal Investigator: Dr. Allen P. Nikora
Phone#: (818)393-1104
Fax#: (818)393-7830
E-mail: Allen.P.Nikora@jpl.nasa.gov
Mail Code: JPL/MS 125-209

Collaborator (separately funded): Dr. John Munson
Managing Member
Cylant Technology
121 Sweet Ave.
Moscow, ID 83843
Phone #: (208)885-5319
E-mail: johnm@cylant.com

INTRODUCTION

Problem Statement

In developing mission software systems, care needs to be taken that the implemented system is sufficiently reliable to carry out the requirements of a multi-year mission. Over the past 30 years, methods to estimate and forecast the reliability of a software system during test and fielded use have been developed, and can be useful in deciding how to allocate scarce testing resources. However, they leave unanswered questions about the system's fault content during earlier development phases, when correction of identified faults requires significantly less effort than during test. Early estimation of a software system's quality is an area of active research in the software reliability engineering community; quite a few techniques have been reported in the literature over the past several years. These techniques, if shown to be applicable to production development efforts, would provide software managers with earlier, more accurate, and more precise control over software quality than they currently have. We propose to pilot the use of one specific set of measurement techniques [Mun98, Niko98], previously developed by the principal investigator and his colleagues, on a JPL software development effort, the Mission Data System (MDS). There are two goals to this task:

1. Validate the findings of the earlier work.
2. Reduce the measurement techniques to practice through automation and standardization.

Justification and Benefits

Successfully piloting these techniques on the MDS will demonstrate their readiness for general use in more accurately and precisely controlling the quality of JPL software systems. Unfortunately, insufficient data exist to accurately estimate the return on investment for this proposal – the difficulty lies in estimating the increase in fault detection efficiency. However, according to [Boehm81], identifying and fixing faults during the coding phase costs approximately half as much as repairing them during development test, and about 1/5 as much as repairing them during acceptance test. Given that repairing faults during acceptance test can take days or weeks at costs in the thousands or tens of thousands of dollars, only a few tens of faults across all projects would need to be discovered during the coding phase rather than during acceptance test to more than pay for this effort.

WORK PLAN DESCRIPTION

Background and Summary

In previous work funded by the NASA IV&V Facility, "IV&V Issues in Achieving High Reliability and Safety in Critical Control System Software", the principal investigator and two colleagues developed methods for predicting and controlling a software system's risk of exposure to residual faults, measuring development process stability, and discriminating between fault-prone components and those which are not fault-prone. Among the findings of the investigators was a relationship between the way the measured structure of a software system changes during its development, and the rate at which faults are inserted into the system. By maintaining a history of a software system's structural evolution during its development, it then becomes possible to estimate its fault content at any point in time, down to the level of individual functions and methods. A follow-on technology-transfer effort, "Assessment and Management of Software Fault Exposure Risk and Quality Throughout the Software Life Cycle", also funded by the NASA IV&V Facility, was to address the practical issues of implementing these measurement techniques on real development efforts (i.e., the MDS). Originally planned for FY99 and FY2000, this task was terminated by the IV&V Facility at the end of FY99 on the grounds that they no longer considered technology transfer to be within their purvue. We propose to complete some of this collaborative work with the MDS, specifically that on piloting the mechanisms for predicting and controlling risk of exposure to residual software faults and estimating test efficiency in a production development environment. Successful completion of this task will bring these measurement and control techniques to a point at which they may be easily used as a tool for more accurately and precisely controlling the quality of JPL mission software systems.

Technical Approach

We propose to perform the following tasks in collaboration with the MDS:

1. **Complete integration of a set of measurement and analysis tools to provide quick feedback during the development effort.** Successively integrate a metrics tool, a tool for computing relative complexity (a measurement of software system structure that is constructed for use as a fault count surrogate, and has been shown to have a high correlation with a system's fault content), a tool for measuring software evolution (the way in which relative complexity changes from build to build), and a profiling tool, into the MDS configuration management and test environments to insure that the complexity measurements for statistical testing, the relative complexity, the system's evolution, and the functional complexity are automatically generated from the code each time the code is modified. Under the cancelled IV&V Facility task, work had started with the MDS configuration management consultant (supplied by TMOD) on integrating these tools into the MDS CM environment. Fault content, the risk of exposure to residual faults, and testing efficiency will be estimated from these measurements. As needed, these tools will be modified to reflect changes to the set of applicable software metrics.
2. **Install and refine previously developed methodology for the correct identification of software faults**. As part of "IV&V Issues in Achieving High Reliability and Safety in Critical Control System Software", an empirical method of producing accurate, consistent, and repeatable

fault counts based on the types of changes made to a software system in response to a reported failure was developed. This empirical taxonomy of software faults will be formalized and extended to cover object-oriented development. Methods to extend existing fault counting methods to earlier phases than implementation will be investigated.

3. **Proof of Concept for Software Risk Assessment**. The number of faults inserted into the MDS during its development will be estimated, as well as the risk of exposure to residual faults based on actual fault counts, structural measurements of the system, and the system's execution profile. Since computing preliminary values for test risk requires knowledge of these systems' execution profiles, the profile tool (see 1 above) will be used to make these estimates.

Dr. John Munson, a co-investigator on the IV&V Facility-funded tasks, will be available on an advisory basis. Dr. Munson developed the measurement tools that will be integrated into the MDS environment, and has previously worked with the PI in developing methods of estimating a software system's fault content during its development. He has agreed to participate in the work described above on an advisory basis. Since Dr. Munson has his own funding, none is requested for him in this proposal.

FLIGHT PROJECT SUPPORT:

Al Sacks, Mission Data System Office Manager, has endorsed this proposal, and will collaborate with the investigators if this proposal is funded. It's his opinion that this work will improve control of the MDS software's fault content and make it more reliable.

GROUP SUPERVISOR APPROVAL:

This proposal has been reviewed and approved by the lead proposer's supervisor, Abdullah Aljabri.

DELIVERABLES/SCHEDULE:

		Due Date	
Activity	**Deliverable**	**FY**	**Quarter**
1	Measurement and Analysis Tools – final versions of tools specifically developed to perform the measurements and analyses reported in the interim technical report. Existing tools that were used to perform the analyses would also be identified and delivered as license agreements allowed.	00	4
2	Fault measurement handbook, to include: • Overview: Estimating fault content, risk of exposure to residual faults, and test efficiency. • Establishing data collection mechanisms. • Tool installation, use, and adaptation instructions. • Interpreting measurement and analysis results. • Measurement standards.	00	4

Activity	Deliverable	Due Date FY	Due Date Quarter
	• Design for a measurement repository		
3	On-going assessment of MDS measurement activities	00	3, 4

ESTIMATED COST:

Cost Item	Cost ($K)
JPL Labor	$42.5
Procurements	$0.0
Travel[1]	$0.6
Services	$0.1
• Document Services	
TOTAL	$43.2

COLLATERAL AND FOLLOW-ON FUNDING:

The U.S. Air Force Operational Test and Evaluation Center (AFOTEC) is funding an effort related to the software metrics study through the remainder of FY2000. The funding will be for the purpose of maintaining and upgrading a previously developed software reliability estimation tool, CASRE. Funding is $25K per fiscal year, through FY2000. The sponsor is also interested in identifying and implementing methods that can be used to estimate and forecast software quality prior to the testing phases. Successful completion of this task would be of interest to the sponsor, and would likely result in additional reimbursable work in this area, the results of which could be applied to JPL development efforts.

The principal investigator, together with the co-investigators on the two IV&V Facility-funded efforts, is developing a proposal in response to NASA Research Announcement NRA 99-OSS-05, "Advanced Cross-Enterprise Technology Development for NASA Missions", to expand on previous work in this area and that proposed here. Their response to the NRA proposes a more detailed examination of relationships between specific types of faults and different types of structural modifications to a system, as well as the conduct of experiments designed to increase our understanding of what types of mistakes are made by developers, and under what conditions, that result in the insertion of faults into a software system that are eventually observed as failures.

Successful completion of this task will result in a set of measurement techniques that are ready for use on mission software systems. It is to be expected that the costs of measurement and analysis would be borne by the development efforts themselves. However, training would be required for those that will make use of these techniques. The PI is searching for NASA funding, starting in FY2001, that would allow the development and piloting of a short course in these measurement techniques.

[1] It is anticipated that for one of the meetings between the PI and the collaborator, the PI will travel.

REFERENCES:

[Boehm81] B. Boehm, Software Engineering Economics, Prentice-Hall, Englewood Cliffs, NJ, 1981, ISBN 0-13-822122-7

[Mun98] J. Munson, A. Nikora, "Estimating Rates of Fault Insertion and Test Effectiveness in Software Systems", invited paper published in the proceedings of the Fourth ISSAT International Conference on Quality and Reliability in Design, Seattle, WA, August 12-14, 1998

[Niko98] A. Nikora, J. Munson, "Determining Fault Insertion Rates For Evolving Software Systems", proceedings of the Ninth International Symposium on Software Reliability Engineering, Paderborn, Germany, November 4-7, 1998